REMARKS

Claims **116-163** are pending in the application.

Claims **116-163** stand rejected.

Claims **116, 118-119, 122-123, 127-128, 130-131, 134, 136-137, 139-140, 143, 145-146, 148-149, 152, 154-155, 157-158, 161** and **163** have been amended.

Rejection of Claims under 35 U.S.C. §103

Claims 116-163 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Bogrett, U.S. Patent No. 6,581,054 (Bogrett) in view of Vandersluis, U.S. Patent No. 6,356,920 (Vandersluis), as per the final Office Action dated June 1, 2007 ("the Office Action"), and maintained in the Advisory Action dated September 6, 2007.

While not conceding that the cited references qualify as prior art, but instead to expedite prosecution, Applicants have chosen to respectfully disagree and traverse the rejection as follows. Applicants reserve the right, for example, in a continuing application, to establish that the cited references, or other references cited now or hereafter, do not qualify as prior art as to an invention embodiment previously, currently, or subsequently claimed.

As an initial matter, Applicants respectfully submit that the particular parts of the cited references that the Examiner has relied upon have not been designated as nearly as practicable, and the pertinence of each reference has not been clearly explained, both as required by 37 C.F.R. § 1.104(c)(2). *See also* MPEP § 706.02(j). The Office Action fails to provide any explanation as to why or how the cited portions of either Bogrett or Vandersluis show, teach or suggest the claim limitations against which they are cited. Nevertheless, Applicants have made every effort to respond to the rejections outlined in the Office Action in a meaningful manner.

Applicants respectfully submit that independent claims 116, 128, 137, 146 and 155 are allowable over Bogrett and Vandersluis, taken alone or in any permissible combination, taking claim 116 as an example, which reads as follows:

> 116. A method comprising:
> generating a set of SQL statements to query a first table and a second table, wherein
> the generating uses a relationship between the first table and the second table to construct the set of SQL statements, and
> the set of SQL statements comprises SQL statements other than a statement that joins the first and second tables;
> querying the first table using the set of SQL statements to produce a result set;
> querying the second table using the set of SQL statements to produce a second result set; and
> joining the result set and the second result set to produce a third result set.

As also noted, Applicants respectfully note that independent claims 128, 137, 146 and 155 recite, among other limitations, substantially similar limitations to those presented above.

By contrast, Bogrett is directed to:

> "A method for generating database queries includes storing a predefined query model for querying a database. An accessible portion of the predefined query model is displayed to a user upon request. User edits to the accessible

portion of the predefined query model are received and used to generate a user-adapted query model. A query is then initiated based on the user-adapted query model." (Bogrett; Abstract)

By further contrast, Vandersluis is directed to:

"A computer system provides the ability to construct and edit a Data Definition File (DDF) containing hierarchically related elements of data, some of which are dynamic in that they must execute in order to produce or retrieve data. A client computer system having knowledge of a DDF appropriate for its uses sends a request to a server, which contains or can retrieve the DDF requested by the client. The request contains parameters used by the server to customize the resulting keyed data file for the client's purposes. Upon receipt of the request, the server copies the DDF into a coupled memory, performs requested parameter substitutions, and executes dynamic elements to produce resulting data elements. The process is repeated recursively for all elements of the hierarchical structure, until no dynamic elements remain, then the resulting keyed data file is returned to the client for its uses. Data elements may be derived from a plurality of sources, and these sources may be combined and manipulated using a plurality of data operations, including relational algebra or structured query language, enabling joins and merges between multiple sources and formats. An Authoring System is provided which assists in the construction and validation of DDFs." (Vandersluis; Abstract)

In order for a claim to be rendered invalid under 35 U.S.C. § 103, the subject matter of the claim as a whole would have to be obvious to a person of ordinary skill in the art at the time the invention was made. *See* 35 U.S.C. § 103(a). This requires: (1) the reference(s) must teach or suggest all of the claim limitations; (2) there must be some teaching, suggestion or motivation to combine references either in the references themselves or in the knowledge of the art; and (3) there must be a reasonable expectation of success. *See* MPEP 2143; MPEP 2143.03; *In re Rouffet*, 149 F.3d 1350, 1355-56 (Fed. Cir. 1998).

As an initial matter, neither Bogrett or Vandersluis, taken alone or in any permissible combination, show, teach or suggest a joinless SQL query that is performed by generating a set of SQL statements to query a first table and a second table, wherein the act of generating uses a relationship between the first table and the second table to construct the set of SQL statements, rather than an SQL statement that joins the first and second tables. Moreover, querying each table, and subsequently joining the results, as is now recited in amended independent claims 117, 128, 137, 146 and 155, in is not shown, taught or suggested in either Bogrett or Vandersluis

The Office Action correctly identifies the former deficiency in Bogrett. Examples of this deficiency include the following passages of Bogrett

> "Next, at step 206, the system administrator generates the predefined query models 56 using the query generator 38. The predefined query models 56 control the elements in a database 20 to which any particular set of users will have access. In addition, the predefined query models 56 restrict the types of queries

> that can be executed and define the maximum computer resources that can be used **to execute the queries and the allowable joins between tables to** prevent run-away or malicious queries." (Bogrett, col. 9, ll. 16-25; Emphasis supplied)

The foregoing portion of Bogrett are nevertheless (erroneously) cited in the Office Action as teaching the claimed invention, notwithstanding the fact that such passages (and Bogrett generally) not only fails to teach the claimed invention, but in fact teaches away from the claimed invention by its use of SQL commands to join multiple tables. As can readily be seen, Bogrett "...execute[s] the queries and the allowable **joins** between tables ..." (Emphasis supplied), and so performs one or more joins the tables in question. This is in direct contravention of the claim language. Further, there is no reason that either reference, having joined the tables, would perform a query on each table (as opposed to the joined tables), and so would have no need of joining multiple results, as is now recited in the amended independent claims.

Vandersluis is relied upon to some how cure the former infirmity, although, given that the use of join statements is explicitly described in Bogrett, Applicants continue to fail to understand how, even if Vandersluis did teach joinless queries (a point which Applicants obviously do not concede), Vandersluis could somehow contravene the explicit use of joins in Bogrett. Such is not the case, of course, as Vandersluis also teaches the use of join statements. For example, in the portion of Vandersluis cited in the Office Action, it is stated that:

> "A list of supported attributes follows:

Attribute Description

EXEC If present, this element is dynamic. This attribute designates the execution type (SQL, ActiveX, Join, Shell, ADO). SQL takes an SQL statement from the Value field and executes it. The target data source of the SQL statement must be specified by a DBDEFINITION element somewhere in the document. The default DBDEFINITION is the first instance in the sequence of parent elements, or any sibling element of a parent. Alternatively, the DBNAME attribute specifies the ID of DBDEFINITON element. ActiveX invokes an ActiveX control to generate data. Join performs a database Join on the children elements. Command executes a shell command to generate data. Shell takes a shell command as the value, executes it using its stdout as data.

JOIN_LABEL **Designates the Type tag to use for resulting**

element sets from a Join

JOIN_TYPE Performs either an Inner or Outer join on the children elements

ROW_TAG The Type tag to use in generated row data for SQL rows or lines of data generated by a Shell Command.

COL_TAG The Type tag to use in generated column data for SQL columns, or token data generated from a Shell Command.

COL_TAGS A list of Type Tags to be used to tag columns returned from an SQL statement or for tokens generated by a Shell Command.

JOIN_KEY The Type tag to look for which defines the element sets to join. When the Value field of the appropriately tagged subelements are equal, the elements are joined to form a single element in the result set

VISIBLE Determines whether the element is visible in the resulting document (default is "YES"). If

> this is "CONTENT_ONLY", the value of the element is transferred to the resulting document, but not the attributes or element type data." (Vandersluis, col. 15, ll. 31-65; Emphasis supplied)

Once again, even in the portion of Vandersluis cited against the claimed invention, the use of one or more join statements is prominently featured. For example, at col. 15, ll. 49-50, Vandersluis clearly states "JOIN_KEY Performs either an Inner or Outer **join** on the children elements" (Emphasis supplied). Applicants are at a complete loss as to how a reference that teaches the use of join statements is even capable of somehow making obvious a claim that specifically calls out the use of a relationship between a first table and a second table to construct a set of SQL statements that include SQL statements <u>other than a statement that joins the first and second tables</u> (thereby specifically excluding the use of join statements).

The Office Action therefore fails to establish the presence of such limitations in Bogrett or Vandersluis, taken alone or in any permissible combination. The Office Action bears the burden of supporting a case of obviousness, including whether the cited references teach or suggest all of the claim limitations. *See* MPEP 706.02(j). For the reasons presented above, neither Bogrett nor Vandersluis, alone or in combination, teach these limitations of claim 116, as well as the remaining independent claims.

Moreover, given that both references must employ join statements, Applicants are once again at a loss as to why (or how) Bogrett and Vandersluis could possibly be combined to arrive at a system that would show, teach or even suggest a joinless SQL query that is performed by generating a set of SQL statements to query a first table and a second table, wherein the act of generating uses a relationship between the first table and the second table to construct the set of SQL statements, rather than an SQL statement that joins the first and second tables.

Further, there is no explanation in the Office Action or the references as to why one of skill in the art, having queried multiple tables through the use of join statements to perform joins on the tables as part of querying those tables, would need (or even be able) to then join the results of the query of the joined table. This is in marked contrast to the claimed invention, which (after performing a joinless query of the tables) joins the results thus obtained to form the final results of the joinless query. Nor is there any teaching or suggestion in the references or the Office Action that the tables each be queried, as in the claimed invention. As will be appreciated, the purpose of a join operation is to join tables together, so that the tables can then be queried *in toto*. Thus, there is no reason for the references to teach multiple queries and the joining of the results of those queries, in a single joinless query operation, as in the claimed invention. Obviously, then, the Office Action cannot establish that such a combination of the teachings of these references would meet with the required success.

For these reasons, Applicants respectfully submit that the Office Action fails to present a *prima facie* case of obviousness with regard to claims 116, 128, 137, 146 and 155, and all claims dependent upon them, and that they are in condition for allowance. Applicants therefore request the Examiner's reconsideration of the rejections to those claims.

CONCLUSION

In view of the amendments and remarks set forth herein, the application is believed to be in condition for allowance and a notice to that effect is solicited. Nonetheless, should any issues remain that might be subject to resolution through a telephonic interview, the Examiner is invited to telephone the undersigned at 512-439-5084.

If any extensions of time under 37 C.F.R. § 1.136(a) are required in order for this submission to be considered timely, Applicant hereby petitions for such extensions. Applicant also hereby authorizes that any fees due for such extensions or any other fee associated with this submission, as specified in 37 C.F.R. § 1.16 or § 1.17, be charged to deposit account 502306.

Respectfully submitted,

Samuel G. Campbell, III
Attorney for Applicants
Reg. No. 42,381
Telephone: (512) 439-5084
Facsimile: (512) 439-5099